Amendments to the Claims

1   Claim 1 (currently amended): A method for programmatically enforcing referential integrity

2   constraints among associations between class instances, comprising steps of:

3   determining, when evaluating a request to set an association end to reflect an association

4   from an instance of a first class to an instance of a second class, whether the association end has

5   a single multiplicity or a many multiplicity;

6   · setting the requested association end; and

7   programmatically modifying an inverse association end of the association to reflect an

8   inverse association from the instance of the second class to the instance of the first class, after

9   disconnecting the inverse association end from an existing instance, if any;

10   wherein an ordering of the setting step and the programmatically modifying step depends

11   on an outcome of the determining step.

Claim 2 (canceled)

1   Claim 3 (currently amended): The method according to Claim [[2]] 1, wherein the ordering of

2   the setting and the programmatically modifying steps for a particular association end where the

3   outcome is that has a the single multiplicity further comprise steps of: comprises

4   disconnecting the inverse association end from an existing instance, if any;

5   performing the programmatically modifying step after performing the disconnecting step;

6   and

7   ———performing the setting step after performing the ~~disconnecting~~ programmatically

8   modifying step.

1   Claim 4 (currently amended): The method according to Claim [[2]] 1, wherein the ordering of

2   the setting and the programmatically modifying steps for a particular association end where the

3   outcome is the ~~that has a~~ many multiplicity further ~~comprise steps of:~~ comprises

4   ———performing the setting step before performing the programmatically modifying step;

5   ———~~disconnecting the inverse association end from an existing instance, if any, after~~

6   ~~performing the setting step; and~~

7   ———~~performing the programmatically modifying step after performing the setting step.~~

1   Claim 5 (original): The method according to Claim 1, further comprising steps of:

2   determining whether the association end or the inverse association end is a primary end of

3   the association; and

4   serializing only the primary end of the association during a serialization operation.

1   Claim 6 (original): The method according to Claim 1, wherein the method is provided as link

2   helper objects.

1   Claim 7 (currently amended): A computer program product for programmatically enforcing

2   referential integrity constraints among associations between class instances, wherein the

Serial No. 09/827,290   -7-   Docket RSW920000173US1

3   computer program product is embodied on one or more computer readable media and comprises:

4   computer-readable program code means for <u>determining, when</u> evaluating a request to set

5   an association end to reflect an association from an instance of a first class to an instance of a

6   second class, <u>whether the association end has a single multiplicity or a many multiplicity</u>;

7   computer-readable program code means for setting the requested association end; and

8   computer-readable program code means for programmatically modifying an inverse

9   association end of the association to reflect an inverse association from the instance of the second

10   class to the instance of the first class<u>, after disconnecting the inverse association end from an</u>

11   <u>existing instance, if any</u>;

12   <u>wherein an ordering of operating the computer-readable program code means for setting</u>

13   <u>and the computer-readable program code means for programmatically modifying depends on an</u>

14   <u>outcome of the computer-readable program code means for determining</u>.

Claim 8 (canceled)

1   Claim 9 (currently amended): The computer program product according to Claim [[8]] <u>7</u>,

2   wherein the <u>ordering of operating the</u> computer-readable program code means for setting and the

3   computer-readable program code means for programmatically modifying for a particular

4   association end <u>where the outcome is the</u> ~~that has a~~ single multiplicity further <u>comprises</u>

5   ~~comprise:~~

6   ~~computer-readable program code means for disconnecting the inverse association end~~

7   ~~from an existing instance, if any;~~

8   ~~computer-readable program code means for operating the computer-readable program~~

9   ~~code means for programmatically modifying after operating the computer-readable program code~~

10   ~~means for disconnecting; and~~

11   computer-readable program code means for operating the computer-readable program

12   code means for setting after operating the computer-readable program code means for

13   ~~disconnecting~~ programmatically modifying; and

14   wherein the ordering of operating the computer-readable program code means for setting

15   and the computer-readable program code means for programmatically modifying for a particular

16   association end ~~that has a~~ where the outcome is the many multiplicity further comprises

17   ~~comprise:~~

18   computer-readable program code means for operating the computer-readable program

19   code means for ~~performing the~~ setting before operating the computer-readable program code

20   means for programmatically modifying;

21   ~~computer-readable program code means for disconnecting the inverse association end~~

22   ~~from an existing instance, if any, after operation of the computer-readable program code means~~

23   ~~for setting; and~~

24   ~~computer-readable program code means for operating the computer-readable program~~

25   ~~code means for programmatically modifying after operating the computer-readable program code~~

26   ~~means for setting.~~

Serial No. 09/827,290   -9-   Docket RSW920000173US1

1   Claim 10 (original): The computer program product according to Claim 7, further comprising:

2   computer-readable program code means for determining whether the association end or

3   the inverse association end is a primary end of the association; and

4   computer-readable program code means for serializing only the primary end of the

5   association during a serialization operation.

1   Claim 11 (currently amended): A system for programmatically enforcing referential integrity

2   constraints among associations between class instances, comprising:

3   means for determining, when evaluating a request to set an association end to reflect an

4   association from an instance of a first class to an instance of a second class, whether the

5   association end has a single multiplicity or a many multiplicity;

6   means for setting the requested association end; and

7   means for programmatically modifying an inverse association end of the association to

8   reflect an inverse association from the instance of the second class to the instance of the first

9   class, after disconnecting the inverse association end from an existing instance, if any;

10   wherein an ordering of operating the means for setting and the means for

11   programmatically modifying depends on an outcome of the means for determining.

Claim 12 (canceled)

1   Claim 13 (currently amended): The system according to Claim [[13]] 11, wherein the ordering of

Serial No. 09/827,290   -10-   Docket RSW920000173US1

2   <u>operating the</u> means for setting and the means for programmatically modifying for a particular

3   association end ~~that has a~~ <u>where the outcome is the</u> single multiplicity further <u>comprises</u>

4   ~~comprise:~~

5   ~~means for disconnecting the inverse association end from an existing instance, if any;~~

6   ~~means for operating the means for programmatically modifying after operating the means~~

7   ~~for disconnecting; and~~

8   means for operating the means for setting after operating the means for ~~disconnecting~~

9   <u>programmatically modifying</u>; and

10   wherein the <u>ordering of operating the</u> means for setting and the means for programmatically

11   modifying for a particular association end ~~that has a~~ <u>where the outcome is the</u> many multiplicity

12   further <u>comprises</u> ~~comprise:~~

13   means for <u>operating the means for</u> ~~performing the~~ setting <u>before operating the means for</u>

14   <u>programmatically modifying</u>;

15   ~~means for disconnecting the inverse association end from an existing instance, if any,~~

16   ~~after operation of the means for setting; and~~

17   ~~means for operating the means for programmatically modifying after operating the means~~

18   ~~for setting.~~

1   Claim 14 (original): The system according to Claim 11, further comprising:

2   means for determining whether the association end or the inverse association end is a

3   primary end of the association; and

Serial No. 09/827,290   -11-   Docket RSW920000173US1

4   means for serializing only the primary end of the association during a serialization

5   operation.

Serial No. 09/827,290   -12-   Docket RSW920000173US1